REPORT DOCUMENTATION PAGE

AFRL-SR-BL-TR-01-

0247

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
		Final Technical Report 01 Oct 97 - 31 Dec 00

4. TITLE AND SUBTITLE
Intelligent Agents for Retrieving, Filtering, and Managing Information

5. FUNDING NUMBERS
F49620-98-1-0046

6. AUTHOR(S)
Craig A. Knoblock

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
University of Southern California
Information Sciences Institute
4676 Admiralty Way
Marina del Rey, CA 90292-6695

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
AFOSR/NM
801 N. Randolph St, Rm 732
Arlington, VA 22203-1977

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
F49620-98-1-0046

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION AVAILABILITY STATEMENT
Approved for public release; distribution unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*
Under this grant we developed the first generation technology for rapidly building instructable agents that automate time -consuming or repetitive tasks. In particular, we solved two critical problems towards achieving the larger vision. First, we developed the machine learning technology to rapidly convert semi-structured data into structured data. This allows web sources to be queries as if they are databases, which is necessary for doing any type of additional filtering, processing, or integration on the data. Second, we developed an agent execution system that makes it easy to define agent plans and to efficiently execute those plans. Our research success under this grant i s best demonstrated by the fact that we have already published a number of journal articles and numerous conference and workshop papers, filed three pa tents on this work, and licensed the resulting technology and software to a startup company that now has 30 employees.

14. SUBJECT TERMS

15. NUMBER OF PAGES
9

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

Accepted 2 APR 01

Final Report

Intelligent Agents for Retrieving, Filtering, and Managing Information

USAF, Air Force Office of Scientific Research
Award Number: F49620-98-1-0046
Period of Performance: 10/01/97 – 12/31/00

Craig A. Knoblock (PI)
University of Southern California
Information Sciences Institute
4676 Admiralty Way
Marina del Rey, CA 90292-6695
Phone: 310-448-8786
Fax: 310-822-0751
Knoblock @isi.edu

Abstract of Technical Progress:

Under this grant we developed the first-generation technology for rapidly building instructable agents that automate time-consuming or repetitive tasks. In particular, we solved two critical problems towards achieving the larger vision. First, we developed the machine learning technology to rapidly convert semistructured data into structured data. This allows web sources to be queries as if they are databases, which is necessary for doing any type of additional filtering, processing, or integration on the data. Second, we developed an agent execution system that makes it easy to define agent plans and to efficiently execute those plans. Our research success under this grant is best demonstrated by the fact that we have already published a number of journal articles and numerous conference and workshop papers, filed three patents on this work, and licensed the resulting technology and software to a startup company that now has 30 employees.

Objectives of the Research Effort:

Recent work on information integration has yielded novel and efficient solutions for gathering data from the World Wide Web. However, there has been little attention given to the problem of providing information management capabilities that closely model how people interact with the web in productive ways - not only collecting information, but monitoring web sites for new or updated data, sending notifications based on the results, building reports, creating local repositories of information, and so on. The overall goal of this research project is to develop the technology and infrastructure for rapidly deploying agents for information management. The key idea is to provide the user with the ability to rapidly specify an information management task by example and then give that task to an agent that is responsible for determining how to efficiently perform the task either on demand or repeatedly at a specified interval. These tasks include retrieving information from one or more data sources, integrating information from these data sources, linking these data sources with planning tools and knowledge bases, filtering and summarizing information, constructing new data sources or briefings, and notifying users when pre-specified conditions hold.

Status of Effort:

At the end of the contract we completed the two key components of an information agent system: wrapper learning for extracting information from online sources and an agent execution system for building efficient information agent plans. These components are described in this section in detail.

Wrapper Learning

In the first year, we developed an approach to rapidly constructing wrappers for Web sites, which allows information agents to extract the data required for information management tasks. To support the wrapper generation we developed a machine learning

approach that allows a user to specify the information to be extracted from a page with only small number of examples. We developed a wrapper induction algorithm that generates extraction rules for semi-structured, Web-based information sources. The algorithm generates extraction rules that are expressed as simple landmark grammars, which are a class of finite automata that is more expressive than the existing extraction languages. The induction method, and an accompanying user interface, enables a user to specify an information extraction task by demonstration. The user simply shows the system what information should be extracted from the example pages. Based on just a few training examples, the system learns extraction rules for sources that can not be wrapped by existing techniques.

In the second year we focused on a limitation of the wrapper learning system, which was that on some of the more complex pages, the system needed many examples in order to learn the correct extraction rules. Since labeling training data is the major bottleneck in all inductive approaches to information extraction, researchers have tried to reduce the burden by using active learning. We created a committee-based active learning algorithm, called SGAL that uses STALKER (the learning algorithm we developed in the first year) to generate 2-member committees of extraction rules. Our approach is similar to *active learning with committees*, except that our committee members are not chosen randomly: the two extraction rules in the committee belong to the most specific and most general borders of the version space. The initial results were promising: we compared SGAL and STALKER on 14 extraction tasks, and the former always does at least as well as the latter, using many fewer examples. More important, SGAL learns 100% accurate rules on four out of the five tasks on which STALKER fails to do so.

In the third year we continued to refine and extend our use of active learning and applied it to the problem of wrapper induction. In particular, we investigated selective sampling algorithms, which asks the user to label only the most informative examples among a given pool of unlabeled examples. Wrapper induction represents a good example of a selective sampling application for two main reasons:

- One can easily create a large pool of unlabeled examples by simply downloading a large number of Web pages. This operation can be done extremely fast and with minimal user effort.
- In order to learn perfect extraction rules, one must browse a large number of unlabeled examples and identify the most informative of them. Selective sampling was designed to replace the human user in this time consuming search process.

In recent work, we proposed the co-testing family of algorithms as a novel approach to selective sampling. Co-testing was inspired by the fact that many real-world learning problems have redundant views. That is, the problem has at least two mutually exclusive sets of features that can be used to learn the target concept.

Co-testing uses the existing redundant views to identify informative examples. In keeping with the selective sampling methodology, co-testing starts with a small set of labeled examples and a pool of unlabeled ones. It first learns a classifier for each view, and then

searches for the unlabeled examples that are classified differently by the learned classifiers. These examples are candidates (contention points), and they represent the candidates from which co-testing selects the next example to be labeled. The various members of the co-testing family use different strategies to select the contention point that the user is asked to label next.

Agent Execution System

In the first year we also developed a plan language and plan execution system for representing and executing the agent plans. Previous work on automatic planning in AI has primarily focused on rather impoverished languages for plans since it is possible to generate these plans automatically from scratch. With more expressive languages, the problem is much harder since it becomes closer to a general automatic programming problem. In our work on information agents, a user provides the plans by example so generating plans from scratch is not required. The types of agent plans that we represent include both loops and conditional branches in the plans. To support these types of plans we have developed a plan execution architecture that can support these more expressive language constructs. In addition, since efficiency is a critical aspect of our work, this language also supports extensive use of parallelism in action execution and pipelining of the data returned from actions. The motivation for these features is that in the Web environment the cost of downloading individual pages often dominates the overall cost of processing. Thus, by exploiting as much parallelism and pipelining as possible, we can most effectively minimize the time required to execute agent plans.

In the second year we completed the implementation and testing of the plan execution system and began work on techniques for automatically optimizing these plans. We developed a complete library of action primitives and used these primitives to build a variety of agent plans. This includes plans that monitor web sites, maintain local data stores, build reports, answer requests on demands, etc. In addition, we began work on the problem of optimizing these plans. Since the goal is to allow users to define plans by example, the given plans may be slow to execute. However, due to the more expressive plan language, the problem of optimizing agent plans is a challenging one. We are currently exploring techniques that build on work from compiler optimization, data flow architectures, and automatic plan merging.

In the third year, we did a complete redesign and implementaion of the Theseus execution platform in an effort to improve the ease of use, scalability, and performance of the system. In particular, we implemented five significant improvements.

- First we simplified the plan language to make plans much easier to hand-write and generate.
- Second, we added fine-grained data streaming. This enables operators to process data at the granularity of a tuple (or groups of tuples), improving the parallelism in a serial flow of operators. Essentially, operators can consume and produce data as it arrives into the system, instead of processing large relations or forcing the plan writer to explicitly iterate through the tuples of a relation.

- Third, we added support for subplans, which provide a way for one plan to call another (including itself), thus enabling a high level of reusability and modularity. Subplans will be able to be called in a parent plan just like any other operator.
- Fourth, we added support for concurrent transactions. This effectively improves the scalability of plan (if it was deployed on a server, multiple concurrent client accesses could be processed). Transactions also enable subplans to be implemented and maintain a resource bound on the number of threads required for execution, something critical to the robust execution of plans exhibiting a high degree of recursion.
- Fifth, we improved the underlying thread model to the Theseus dataflow execution unit: our goal here is to further decentralize execution, maintain pools of operator "worker" threads at runtime (as opposed to on-demand instantiation), and to optimize the queuing behavior of consumer operators.
- Finally, we extended to original relational representation to an XML-based representation, which allows the system to operate on more complicated object structures. The set of operations in Theseus is separate from the execution platform, so the original relational representation is still available for cases where it is more appropriate.

Publications:

Jose Luis Ambite and Craig A. Knoblock.
Flexible and scalable query planning in distributed and heterogeneous environments.
In *Proceedings of the Fourth International Conference on Artificial Intelligence Planning Systems*, Pittsburgh, PA, 1998.

Naveen Ashish, Craig A. Knoblock, and Cyrus Shahabi.
Semantic caching for information integration.
In *Proceedings of the AAAI'98 Workshop on AI and Information Integration*, Madison, WI, 1998.

Craig A. Knoblock, Steven Minton, Jose Luis Ambite, Naveen Ashish, Pragnesh Jay Modi, Ion Muslea, Andrew G. Philpot, and Sheila Tejada.
Modeling web sources for information integration.
In *Proceedings of the Fifteenth National Conference on Artificial Intelligence*, Madison, WI, 1998.

Ion Muslea, Steven Minton, and Craig A. Knoblock.
Wrapper induction for semistructured, web-based information sources.
In *Proceedings of the Conference on Automated Learning and Discovery Workshop on Learning from Text and the Web*, Pittsburgh, PA, 1998.

Ion Muslea, Steve Minton, and Craig A. Knoblock.
A Hierarchical Approach to Wrapper Induction.
Proceedings of the Third International Conference on Autonomous Agents, Seattle, WA, 1999.

Naveen Ashish, Craig A. Knoblock and Cyrus Shahabi
Selectively Materializing Data in Mediators by Analyzing User Queries.
Proceedings of the Fourth IFCIS Conference on Cooperative Information Systems, Edinburgh, Scotland, 1999.

Jose Luis Ambite, Craig A. Knoblock, and Steven Minton.
Learning plan rewriting rules.
In Proceedings of the Fifth International Conference on Artificial Intelligence Planning and Scheduling Systems, Breckenridge, CO, 2000.

Greg Barish, Daniel DiPasquo, Craig A. Knoblock, and Steven Minton.
Dataflow plan execution for software agents.
In Proceedings of the Fourth International Conference on Autonomous Agents (Agents-2000) Poster Session , Barcelona, Spain, 2000.

Greg Barish, Daniel DiPasquo, Craig A. Knoblock, and Steven Minton.
A dataflow approach to agent-based information management.
In Proceedings of the 2000 International Conference on Artificial Intelligence (IC-AI 2000) , Las Vegas, NV, 2000.

Greg Barish, Craig A. Knoblock, Yi-Shin Chen, Steven Minton, Andrew Philpot, and Cyrus Shahabi.
The theaterloc virtual application.
In Proceedings of Twelfth Annual Conference on Innovative Applications of Artificial Intelligence (IAAI-2000) , Austin, Texas, 2000.

Ion Muslea, Steven Minton, and Craig A. Knoblock.
Selective sampling with redundant views.
In Proceedings of the 17th National Conference on Artificial Intelligence (AAAI-2000) , 2000.

Jose Luis Ambite and Craig A. Knoblock.
Flexible and scalable cost-based query planning in mediators:
A transformational approach.
Artificial Intelligence Journal, 118(1-2):115--161, April 2000.

Ion Muslea, Steven Minton, and Craig A. Knoblock.
Hierarchical wrapper induction for semistructured information sources.
To appear in the Journal of Autonomous Agents and Multi-Agent Systems, Special Issue on ``Best of Agents'99'', 4(1/2), March, 2001.

Accurately and reliably extracting data from the web: A machine learning approach,
Craig A. Knoblock, Kristina Lerman, Steven Minton, and Ion Muslea.
Data Engineering Bulletin, 23(4), December, 2000.

Jose Luis Ambite, Craig A. Knoblock, Ion Muslea, and Andrew Philpot.

Compiling source descriptions for efficient and flexible information integration.
To appear in the Journal of Intelligent Information Systems, Forthcoming.

Selectively materializing data in mediators by analyzing user queries,
Naveen Ashish, Craig A. Knoblock, and Cyrus Shahabi.
International Journal of Cooperative Information Systems, Forthcoming.

Craig A. Knoblock, Steven Minton, Jose Luis Ambite, Naveen Ashish, Ion Muslea, Andrew G. Philpot, , and Sheila Tejada.
The ariadne approach to web-based information integration.
To appear in the International the Journal on Intelligent Cooperative Information Systems (IJCIS) Special Issue on Intelligent Information Agents: Theory and Applications, Forthcoming.

List of Personnel Associated with the Research Effort:

Craig Knoblock, PI
Steven Minton, Co-Project Leader (transitioned to Fetch.com in July)
Jose Luis Ambite, Senior Research Scientist
Dan Dipasquo, Research Scientist
Jay Modi, Research Scientist
Dan Rosenberry, Research Scientist
Naveen Ashish, Graduate Research Assistant
Greg Barish, Graduate Research Assistant
Ion Muslea, Graduate Research Assistant

Significant Events:

- Steve Minton became a AAAI Fellow, 1998.
- Craig Knoblock was promoted to Research Associate Professor, 1999.
- Ion Muslea received a best student paper award at the Agents-99 conference.
- Steve Minton left USC in July 2000 to become the CTO of Fetch.com, which has licensed the technology developed under this grant and is commercializing both the wrapper learning and agent execution platform.

Interactions/Transitions:

- Knoblock & Minton -- Participated in the AFOSR annual meeting, Sept, 1997
- Minton – Gave invited keynote address at Constraint Programming Conference, October 1997
- Knoblock, Minton, Modi, Ambite, Ashish & Muslea -- Attended AAAI-98 and Knoblock presented their joint paper, July, 1998
- Knoblock, Minton, Ambite, Ashish, & Muslea -- Participated in the Workshop on AI and Information Integration. Knoblock was an organizer of the workshop, Ashish &

Muslea both presented papers, and Knoblock & Minton both participated on panels, July 1998

- Knoblock & Ambite -- Attended AI Planning Systems Conference and Ambite presented their joint paper, July, 1998
- Knoblock & Ambite -- Participated in the workshop on Planning as Combinatorial Search and they both participated on panels, July, 1998
- Minton & Muslea -- Attended Conald-98 and Muslea presented their joint paper, July, 1998
- Minton – Gave an invited talk at the Central Intelligence Agency on Information Agents for Web Sources, Oct, 1998.
- Knoblock and Minton – Participated in the DARPA COABS Meeting in Las Vegas, NV, Jan, 1999.
- Knoblock & Minton -- Participated in the AFOSR annual meeting, Feb, 1999.
- Knoblock and Minton – Participated in the DARPA COABS Meeting in Northampton, MA, June, 1999.
- Knoblock and Minton – Participated in the DARPA COABS Meeting in Atlanta, GA, Jan, 2000.
- Knoblock – Hosted and participated in the AFOSR annual meeting, March, 2000.
- Knoblock – Participated in the DARPA Active Templates Meeting in Breckenridge, CO, April 2000.
- Knoblock– Participated in the DARPA COABS Meeting in Boston, MA, August, 2000.
- Knoblock – Visited AFRL and presented his work on various Air Force related projects, July, 2000.

Papers Presented:

- Ion Muslea, Steven Minton, and Craig Knoblock, A Hierarchical Approach to Wrapper Induction, Third International Conference on Autonomous Agents, Seattle, WA, 1999. Ion Muslea presented the paper.
- Craig Knoblock and Steven Minton, Building Agents for Internet-based Supply Chain Integration, Workshop on Agent-based Decision Support for Managing the Internet Enabled Supply Chain, Seattle, WA, May, 1999. Craig Knoblock presented the paper.
- Craig Knoblock, Building Agents for Integrating and Managing Data from the Web, The Boeing Corporation, June, 1999.
- Greg Barish, Craig A. Knoblock, Yi-Shin Chen, Steven Minton, Andrew Philpot, and Cyrus Shahabi, TheaterLoc: A Case-Study in Information Integration, IJCAI Workshop on Intelligent Information Integration, July, 1999. Craig Knoblock presented the paper and was an organizer of the workshop.
- Ion Muslea, Extraction Patterns for Information Extraction Tasks: A Survey, AAAI Workshop on Machine Learning for Information Extraction, Orlando, FL, July, 1999.
- Ion Muslea, Steve Minton, Craig Knoblock, Active Learning for Hierarchical Wrapper Induction, Student poster at AAAI, July, 1999. Ion Muslea presented the poster.

- Jose Luis Ambite, Craig A. Knoblock, and Steven Minton.
  Learning plan rewriting rules. Conference on Artificial Intelligence Planning and Scheduling Systems, April 2000. Jose Luis Ambite presented the paper.
- Greg Barish, Daniel DiPasquo, Craig A. Knoblock, and Steven Minton.
  Dataflow plan execution for software agents. Fourth International Conference on Autonomous Agents (Agents-2000), June, 2000. Greg Barished presented the poster.
- Greg Barish, Daniel DiPasquo, Craig A. Knoblock, and Steven Minton.
  A dataflow approach to agent-based information management.
  International Conference on Artificial Intelligence (IC-AI 2000) , June, 2000. Greg Barish presented the paper.
- Greg Barish, Craig A. Knoblock, Yi-Shin Chen, Steven Minton, Andrew Philpot, and Cyrus Shahabi. The theaterloc virtual application. Twelfth Annual Conference on Innovative Applications of Artificial Intelligence (IAAI-2000) , August 2000. Greg Barish presented the paper.
- Ion Muslea, Steven Minton, and Craig A. Knoblock.
  Selective sampling with redundant views.
  National Conference on Artificial Intelligence (AAAI-2000) , August 2000.

Additional Information:

- Filed a Provisional Patent Application entitled Information Agent System (Theseus), 11/4/99.
- Filed a Provisional Patent Application entitied Co-Testing, 4/7/00.
- Filed a Utility Patent Application entitled Hierarchical Rule Induction for Extracting Data from Semistructured Documents, 6/2/00.